Amendments In the Claims

Please amend Claims 1, 10, and 16-18 as follows:

1.   (**Currently Amended**) A method for transforming one or more lists for a data communications system into a single list, each list of the one or more lists including a plurality of entries, the method comprising:

removing non-terminating entries from the plurality of entries in the one or more
   lists, the removing each non-terminating entry removing all but a last non-
   terminating entry in any of the one or more lists; ~~and~~

determining whether one or more entries of the plurality of entries provides
   for an impossible action, wherein
   each entry of the plurality of entries comprises an associated
      condition, and
   said determining comprises
      selecting a first entry of the plurality of entries, and
      for each entry of the plurality of entries except the first entry,
         determining whether the condition associated with the
            entry can be satisfied if a first condition
            associated with the first entry is satisfied, and
            identifying the entry as providing for an
            impossible action if the condition associated with
            the entry cannot be satisfied; and

eliminating from the plurality of entries **the** one or more entries**, if any,** that
   provide for **an** ~~one or more~~ impossible action[[s]], wherein[[:]]
   the removing of non-terminating entries and the eliminating of **the** one or
      more entries that provide for **an** impossible action[[s]], if any,
      produce a single list preserving tracing of the entries in the single
      list to the plurality of entries.

2.   (Original) The method of Claim 1 wherein each of the plurality of entries provides an indication of a source of the entry with an action code to create a pairing.

3.   (Original) The method of Claim 1 wherein the preserving tracing permits determining statistical parameters of the data communication system.

4.   (Original) The method of Claim 3 wherein the statistical parameters include counts of matched entries.

5.   (Original) The method of Claim 1 wherein the removing and the eliminating is performed at a network element of the data communication system.

6.   (Original) The method of Claim 1 wherein the list of entries is an Action Control List (ACL) and wherein each entry is an Action Control Entry (ACE).

7.   (Original) The method of Claim 1 wherein the one or more lists of data are action control lists (ACLs), the method further comprising:
> combining at least a first and a second ACL by combining each non-terminating
>> entry in a first ACL with each entry in a second ACL; and
> repeating the combining recursively to a third or more ACLs, if present, until each
>> ACL is collapsed into the first ACL to create the single list.

8.   (Original) The method of Claim 7 wherein the single list holds a plurality of Action Control Entries (ACEs) that are codeable into a first match engine capable of computing a large number of Boolean expressions in parallel and returning an index of first matching ACEs.

9.   (Original) The method of Claim 8 wherein the first match engine is implemented as one of a ternary content addressable memory (TCAM) and a hardware device capable of computing a large number of Boolean expressions in parallel and returning an index of first matching ACEs.

Serial No.: 10/001,615

10.   (**Currently Amended**) A data routing system to administer entries, the data routing system comprising:

a network element configured to receive a plurality of Action Control Lists

(ACLs) organized to hold a plurality of Action Control Entries (ACEs);

a processor configured to

receive the plurality of ACLs holding the plurality of ACEs, ~~, the~~

~~processor adapted to:~~

remove any non-terminating entries from the plurality of ACEs in the

plurality of ACLs, wherein the removal of each non-terminating

entry removes all but a last non-terminating entry in any of the

ACLs, ~~; and~~

determine whether one or more ACEs of the plurality of ACEs

provides for an impossible action, wherein

each ACE of the plurality of ACEs comprises an associated

condition, and

the processor is further configured to

select a first ACE from the plurality of ACEs, and

for each ACE of the plurality of ACEs except the first

ACE, determine whether the condition

associated with the ACE can be satisfied if a first

condition associated with the first ACE is

satisfied, and identify the ACE as providing for

an impossible action if the condition associated

with the ACE cannot be satisified; and

eliminate from the **plurality of** ACEs **the** one or more ACEs that provide

for **an** ~~one or more~~ impossible action[[s]] if present, wherein the

removal of non-terminating entries and the elimination of one or

more ACEs that provide for **an** impossible action[[s]] produce<u>s</u> a

single list with entries, the single list configured to preserve tracing

of the entries in the single list to the plurality of ACEs.

11.   (Original) The data routing system of Claim 10 further comprising:

a hardware device coupled to receive the single list with entries, the hardware

        device being a parallel-first match engine.

12.   (Original) The data routing system of Claim 11 wherein the hardware

device is one of a content addressable memory and a ternary content addressable

memory.

13.   (Original) The data routing system of Claim 11 wherein the single list

with entries is coded for presentation to the hardware device.

14.   (Original) The data routing system of Claim 10 wherein the one or more

ACEs provide for a plurality of actions.

15.   (Original) The data routing system of Claim 10 wherein the one or more

ACEs provide for one or more of:

        encryption and decryption, web caching, tunneling, redirection to a predetermined

                router interface, redirection to a separate processor or linecard for one or

                more of the encryption, decryption, web caching, and tunneling.

16.   **(Currently Amended)** A computer system comprising:

a processor; and

a memory, the memory including instructions[[,]] **executable by** the processor ~~for~~

        ~~executing the instructions~~, the instructions including encoding

        instructions for one or more lists, each list including a plurality of entries,

        the encoding instructions **comprising** ~~including:~~

        skip entry removal instructions for removing non-terminating entries from

                the plurality of entries in the one or more lists, the removing each

                non-terminating entry removing all but a last non-terminating entry

                in any of the one or more lists, **;** ~~; and~~

Serial No.: 10/001,615

<u>impossibility entry determination instructions for determining</u>

 <u>whether one or more entries of the plurality of entries provides</u>

 <u>for an impossible action, wherein</u>

 <u>each entry of the plurality of entries comprises an associated</u>

  <u>condition, and</u>

 <u>the impossibility entry determination instructions further</u>

  <u>comprise</u>

  <u>selection instructions for selecting a first entry in the</u>

   <u>plurality of entries, and</u>

  <u>impossibility identification instructions for identifying</u>

   <u>an impossible action for each entry of the</u>

   <u>plurality of entries except the first entry further</u>

   <u>comprising</u>

   <u>a first set of instructions for determining</u>

    <u>whether the condition associated with the</u>

    <u>entry can be satisfied if a first condition</u>

    <u>associated with the first entry is satisfied,</u>

    <u>and</u>

   <u>a second set of instructions for identifying the</u>

    <u>entry as providing for an impossible</u>

    <u>action if the condition associated with the</u>

    <u>entry cannot be satisfied, and</u>

impossibility entry elimination instructions for removing from the

 plurality of entries **the** one or more entries that provide for **an** ~~one~~

 ~~or more~~ impossible action[[s]], **if any,** wherein the removal of

 non-terminating entries and the removal of **the** one or more entries

 that provide for **an** impossible action[[s]] produce a single list

 preserving tracing of the entries in the single list to the plurality of

 entries.

17.   (**Currently Amended**) A computer program product, ~~the computer program product~~ comprising:

> signal bearing media bearing digital information adapted to include programming, the digital information including:

>> a block configured to remove non-terminating entries from **a** ~~the~~ plurality of entries in ~~the~~ one or more lists, the **block further configured to** ~~removing each non-terminating entry removing~~ **remove** all but a last non-terminating entry in any of the one or more lists; ~~and~~

>> **a block configured to determine whether one or more entries of the plurality of entries provides for an impossible action, wherein each entry of the plurality of entries comprises an associated condition, and**

>>> **the block is further configured to**

>>>> **select a first entry of the plurality of entries, and**

>>>> **for each entry of the plurality of entries,**

>>>>> **determine whether the condition associated with the entry can be satisfied if a first condition associated with the first entry is satisified, and identify the entry as providing for an impossible action if the condition associated with the entry cannot be satisified; and**

>> a block configured to eliminate from the plurality of entries **the** one or more entries that provide for **an** ~~one or more~~ impossible action[[s]], wherein[[:]] the removal of non-terminating entries and the elimination of one or more entries that provide for **an** impossible action[[s]] produce a single list preserving tracing of the entries in the single list to the plurality of entries.

18.   (Currently Amended) A network element configured to transform one or more lists for a network, each list including one or more entries, the network element comprising:

means for removing non-terminating entries from the one or more entries in the one or more lists, the removing each non-terminating entry removing all but a last non-terminating entry in any of the one or more lists; ~~and~~

means for determining whether a subset of entries of the one or more of entries provide for an impossible action, wherein

each entry of the one or more entries comprises an associated condition, and

said means for determining comprises

means for selecting a first entry of the one or more entries, and for each entry of the one or more entries except the first entry,

means for determining whether the condition associated with the entry can be satisified if a first condition associated with the first entry is satisfied, and

means for identifying the entry as a member of the subset of entries providing for an impossible action if the condition associated with the entry cannot be satisfied; and

means for eliminating from the one or more entries **the subset of entries** ~~each entry~~ that provide[[s]] for **an** ~~one or more~~ impossible action[[s]], wherein the means for removing of non-terminating entries and the means for eliminating ~~each entry~~ **the subset of entries** that provide[[s]] for **an** impossible action[[s]] provide a means for producing a single list preserving tracing of the entries in the single list to the one or more entries.